

4 インストール説明書

4.1 発注側 EDI サーバ設定

ここでは、発注側 EDI サーバ機能のインストール方法、および、使用環境設定方法について説明します。

- ・プログラム導入 4.1.1 参照
- ・使用環境設定 4.1.2 参照
- ・社内システムとのインタフェース 4.1.3 参照

4 . 1 . 1 プログラム導入

ここでは、発注側 EDI サーバ機能のパッケージ内容、インストール手順を説明します。

- ・パッケージ内容 (1)参照
- ・インストール手順 (2)参照

(1) パッケージ内容

EDI サーバプログラムは以下のパッケージにより構成されます。

名称	内容説明	インストール先
認証ドキュメント用ディレクトリ	ドキュメントを保管します。	/home/wg71/secure/htdocs/
認証サーブレット関連ドキュメント用ディレクトリ	サーブレット関連ドキュメントを保管します。	/home/wg71/secure/htdocs/servlets/
認証ダウンロード用ディレクトリ	ダウンロードファイルを保管します。	/home/wg71/secure/htdocs/download/
認証アップロード用ディレクトリ	アップロードファイルを保管します。	/home/wg71/secure/htdocs/upload/
FAX 督促用ディレクトリ	FAX 督促データを保管します。	/home/wg71/secure/htdocs/reminder/
認証 CGI ディレクトリ	CGI を保管します。	/home/wg71/secure/cgi-bin/
認証サーブレットディレクトリ	サーブレットを保管します。	/home/wg71/secure/servlets/
Java クラスディレクトリ	Java クラスを保管します。	/home/wg71/secure/servlets/Servlet.com
掲示板用ディレクトリ	掲示板データを保管します。	/home/wg71/secure/htdocs/board/
認証ログディレクトリ	ログファイルを保管します。	/home/wg71/secure/logs/
管理者ドキュメントディレクトリ	ドキュメントを保管します。	/home/wg71/system/htdocs/
エラーログ用ディレクトリ	エラーログファイルを保管します。	/home/wg71/system/htdocs/errorlog
実行ログ用ディレクトリ	実行ログファイルを保管します。	/home/wg71/system/htdocs/execlog
管理者 CGI ディレクトリ	CGI を保管します。	/home/wg71/system/cgi-bin/
管理者サーブレットディレクトリ	サーブレットを保管します。	/home/wg71/system/servlets/
管理者ログディレクトリ	ログファイルを保管します。	/home/wg71/system/logs/
メーカ情報、クライアント情報関連テンプレート用ディレクトリ	メーカ情報、クライアント情報関連のテンプレートを保管します。	/home/wg71/template/admedi/
発注側データ表示関連テンプレート用ディレクトリ	発注側データ表示関連テンプレートを保管します。	/home/wg71/template/viewedi/

アクセス管理用一時ファイル用ディレクトリ	アクセス管理用一時ファイルを保管します。	/home/wg71/system/outer/session/
発注者ドキュメントディレクトリ	ドキュメントを保管します。	/home/wg71/customer/htdocs
発注者 CGI ディレクトリ	CGI を保管します。	/home/wg71/customer/cgi-bin/
発注者サーブレットディレクトリ	サーブレットを保管する。	/home/wg71/customer/servlets/
発注者ログディレクトリ	ログファイルを保管します。	/home/wg71/customer/logs/
受発注マスタディレクトリ	受発注マスタを保管します。	/home/wg71/edi/master/
受発注マスタ通信テスト用	通信テスト用データを保管します。	/home/wg71/edi/master/testdata/
社内システム取込用ディレクトリ	社内システムから送られるファイルを保管します。	/home/wg71/edi/inner/input/
社内システム送込一時ディレクトリ	社内システム送込用の一時データを保管します。	/home/wg71/edi/inner/temp/
社内システム送込用ディレクトリ	社内システムへ送られるファイルを保管します。	/home/wg71/edi/inner/output/
受発注マスタ登録作業ディレクトリ	受発注マスタ登録用一時データを保管します。	/home/wg71/edi/temp/
受注側クライアントからのアップロード用ディレクトリ	受注クライアントからアップロードされたファイルを保管します。	/home/wg71/edi/outer/
入力用データディレクトリ	入力用データを保管します。	/home/wg71/edi/outer/input/
回答ファイルディレクトリ	回答ファイルを保管します。	/home/wg71/edi/outer/input/answer/
回答履歴用ディレクトリ	回答履歴ファイルを保管します。	/home/wg71/edi/outer/input/answerlog/
メーカー情報変更データ用ディレクトリ	メーカー情報変更データを保管します。	/home/wg71/edi/outer/input/makerinfo/
出力用データディレクトリ	出力用データを保管します。	/home/wg71/edi/outer/output
送信データ用ディレクトリ	送信データを保管します。	/home/wg71/edi/outer/output/send/
送信履歴用ディレクトリ	送信履歴ファイルを保管します。	/home/wg71/edi/outer/output/sendlog/
公開メソッドの作業用ディレクトリ	公開メソッドの作業データを保管します。	/home/wg71/edi/outer/session/
履歴ファイル用ディレクトリ	履歴ファイルを保管します。	/home/wg71/edi/logs/

メーカー情報用ディレクトリ	メーカー情報ファイルを保管します。	/home/wg71/maker/information/
セキュリティ情報用ディレクトリ	セキュリティ情報を保管します。	/home/wg71/client/security/
環境設定ファイル用ディレクトリ	環境設定ファイルを保管します。	/home/wg71/env/
定時実行処理設定ディレクトリ	定時実行処理設定ファイルを保管します。	/home/wg71/env/regTime/
定時実行 P G M パス	定時実行しますプログラムを保管します。	/home/wg71/pgm/regTime
DTD 用ディレクトリ	DTD を保管します。	/home/wg71/dtd/
サーバプログラム (モジュール)	サーバプログラムを保管します。	/home/wg71/pgm/
プログラム管理用ディレクトリ	プログラム管理用データを保管します。	/home/wg71/progman/
プログラム管理用一時ディレクトリ	プログラム管理用一時データを保管します。	/home/wg71/progman/temp/

(2) インストール手順

以下にプログラムの導入手順を記述します。

なお、導入前の環境設定については 4 . 1 . 2 項の使用環境設定により、設定します。

(A) インストール作業

以下の媒体を準備し、プログラムをインストールします。

防衛調達 C A L S 航空機業界標準 EDI システム発注側 E D I サーバ用 C D
- R O M

本手順は、基本的な UNIX コマンドを熟知しているものとして記述してあります。又、本文中での表記規則は、以下の通りである。

表記規則	内容
#	サーバ上でのコマンド入力 (ルートユーザ)
%	サーバ上でのコマンド入力 (一般ユーザ)

(B) プログラムの作動確認作業

(A) の導入作業後、正常に動作することを確認します。

4 . 1 . 1 プログラム 導入	(2) 導入手順	1/2
<p>(A) インストール</p> <p>(a) サーバ用プログラムのインストール</p> <p>(ア) ルートユーザでログインします。</p> <p>(イ) C D - R O M をマウントします。</p> <pre>#mount /dev/cdrom /mnt/cdrom</pre> <p>(ウ) C D - R O M よりインストール用コマンド及びインストール用アーカイブファイルを任意のディレクトリにコピーします。</p> <pre>#cd /tmp/ #cp /mnt/cdrom/* ./</pre> <p>(エ) インストールコマンドファイルを実行します。</p> <pre>#inst_wg71</pre> <p>(オ) C D - R O M のマウントを解除します。</p> <pre>#umount /mnt/cdrom</pre>		

4 . 1 . 1 プログラム 導入	(2) 導入手順	2/2
<p>(B) プログラムの作動確認作業</p> <p>(a) サーバ運用管理プログラム</p> <p>システム管理メニューを立ち上げ、全てのリンクをクリックし、サブメニューが正常に表示されることを確認します。</p> <p>アドレス : http://サーバの URL 名:27163/servlets/admedi.RootMenu</p> <p>例 : http://mhi170.ctc.ne.jp:27163/servlets/admedi.RootMenu</p> <p>(b) 発注者クライアントプログラム</p> <p>発注側クライアントメニューを立ち上げ、全てのリンクをクリックし、サブメニューが正常に表示されることを確認します。</p> <p>アドレス : http://サーバの URL 名:31682/servlets/viewedi.ViewerMenu</p> <p>例 : http://mhi170.ctc.ne.jp:31682/servlets/viewedi.ViewerMenu</p>		

4.1.2 使用環境設定

ここでは、発注側 EDI サーバ機能の使用環境設定方法を説明します。

- ・ プログラム環境設定 (1)参照
- ・ ベースソフトウェア環境設定 (2)参照
- ・ フリーソフトウェアのインストール及び設定 (3)参照
- ・ インターネットへの接続確認 (4)参照
- ・ フリーソフトウェアの稼働確認 (5)参照

以下の手順は、基本的な UNIX コマンドを熟知しているものとして記述してあります。

本文中での表記規則は、以下の通りです。

表記規則	内容
#	サーバ上でのコマンド入力（ルートユーザ）
%	サーバ上でのコマンド入力（一般ユーザ）

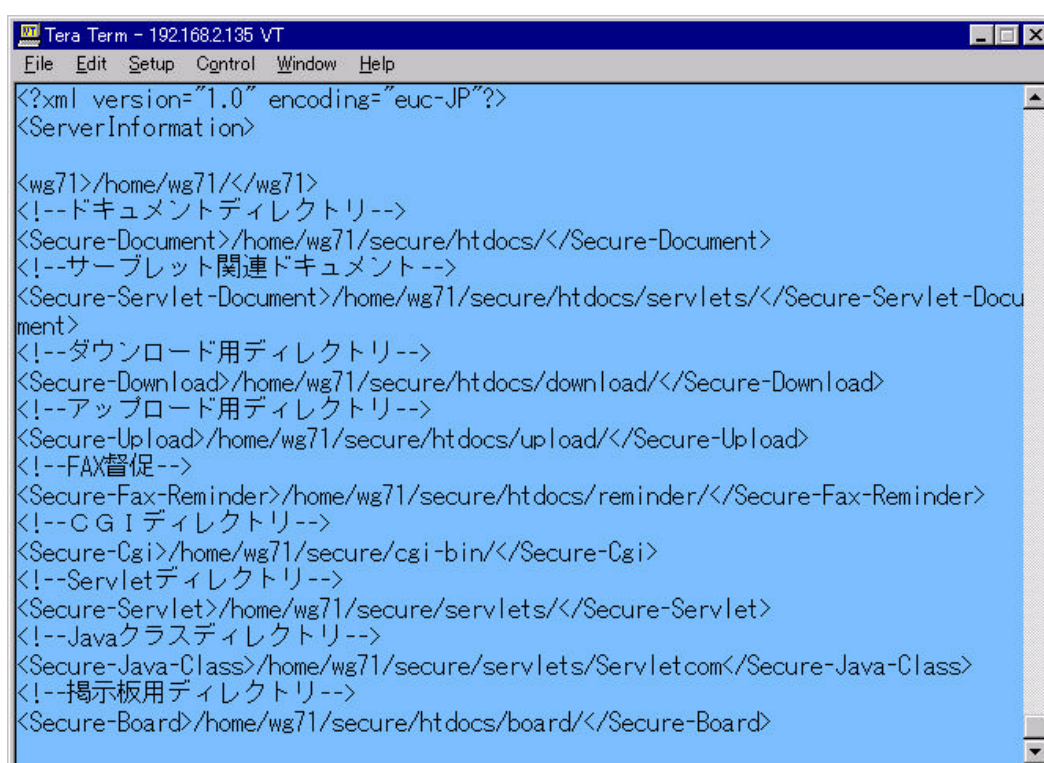
(1) プログラム環境設定

変更の必要がある場合のみ、プログラムの環境を定義しているファイルをエディタにて編集します。通常は変更の必要はありません。

例)

UNIX 標準 vi エディタでプログラムの環境を定義しているファイルを開く。
プログラム環境とは、プログラムが参照している、ファイルやディレクトリが
定義されているファイルです。

```
# vi /home/wg71/env/edi_server_env_.xml
```



```
<?xml version="1.0" encoding="euc-JP"?>
<ServerInformation>

<wg71>/home/wg71/</wg71>
<!--ドキュメントディレクトリ-->
<Secure-Document>/home/wg71/secure/htdocs/</Secure-Document>
<!--サーブレット関連ドキュメント-->
<Secure-Servlet-Document>/home/wg71/secure/htdocs/servlets/</Secure-Servlet-Document>
<!--ダウンロード用ディレクトリ-->
<Secure-Download>/home/wg71/secure/htdocs/download/</Secure-Download>
<!--アップロード用ディレクトリ-->
<Secure-Upload>/home/wg71/secure/htdocs/upload/</Secure-Upload>
<!--FAX督促-->
<Secure-Fax-Reminder>/home/wg71/secure/htdocs/reminder/</Secure-Fax-Reminder>
<!--CGIディレクトリ-->
<Secure-Cgi>/home/wg71/secure/cgi-bin/</Secure-Cgi>
<!--Servletディレクトリ-->
<Secure-Servlet>/home/wg71/secure/servlets/</Secure-Servlet>
<!--Javaクラスディレクトリ-->
<Secure-Java-Class>/home/wg71/secure/servlets/Servletcom</Secure-Java-Class>
<!--掲示板用ディレクトリ-->
<Secure-Board>/home/wg71/secure/htdocs/board/</Secure-Board>
```

ファイルの内容を変更します。以下に変更例を示します。

```
<!--社内システムからサーバへ-->      ディレクトリの説明です。
<Edi-Inner-Input>/home/wg71/edi/inner/input/</Edi-Inner-Input>
```

上記の網掛けの部分の文字を変更したいディレクトリに変更します。

例

```
<Edi-Inner-Input>/home/wg71/edi/customer/in/</Edi-Inner-Input>
```

(2) ベースソフトウェア環境設定

ベースソフトウェアであるOS (Vine Linux 1.1) に関する環境設定内容の詳細および、Webサーバのユーザである webuser に関する環境設定について記述します。

4 . 1 . 2 使用環境設定	(2) ベースソフトウェア環境設定	1/6
------------------	---------------------	-----

(A) OSのインストール確認および環境設定

(a) OSの確認

以下のOSがインストールされていることを確認します。

Vine LINUX 1.1

起動画面に以下のログインプロンプトが表示されている

画面表示例：

```
Vine Linux 1.1CR (Rheingau)
Kernel 2.0.36 on an i686
login:
```

(b) パーティションの設定確認

以下にパーティション設定の基本設定（推奨）に近い数値が各パーティションに割り当てられていることを確認します。。尚、ハードディスクが2つ以上あるサーバの場合はハードディスクの1つを「/home」に割り当てることを推奨します。

マウント位置	サイズ	拡張	種類
(スワップ)	1 2 7 M	拡張可	Linux Swap
(スワップ)	1 2 7 M	拡張可	Linux Swap
/	1 0 0 M以上	拡張可	Linux Native
/usr	1 0 0 0 M以上	拡張可	Linux Native
/var	2 0 0 M以上	拡張可	Linux Native
/home	上記設定後に残ったサイズ	拡張可	Linux Native

MHI での設定例を以下に示します

マウント位置	デバイス名	サイズ	拡張	種類
(スワップ)	sda	1 2 7 M	拡張可	Linux Swap
(スワップ)	sda	1 2 7 M	拡張可	Linux Swap
/	sda	1 0 0 0 M	拡張可	Linux Native
/usr	sda	2 0 0 0 M	拡張可	Linux Native
/var	sda	1 0 0 0 M	拡張可	Linux Native
/home	sda	1 7 3 5 7 M	拡張可	Linux Native

(c) コンポーネントの選択の確認

インストール時にコンポーネントの選択メニューより以下のコンポーネントを追加選択されていることを確認します。

- ・ LAN 接続
- ・ インターネットサーバ (DNS/WWW/News)

(d) グループの作成

本システムが稼働する為に必要なグループをルートユーザにログインし、作成します。なお、グループの作成は、システムコマンドまたは、V i n e L i n u x の管理ツールより実施します。

以下、コマンドでのグループの作成方法を示します。

```
# groupadd -g グループ ID グループ名
```

例) admusers の場合

```
# groupadd -g 500 admusers
```

本システムに必要なグループを以下に示します。

グループ名	グループ ID	内容
admusers	500	管理者用グループ
webuser	501	WWWサーバ用グループ

(e) ユーザの作成

本システムが稼働する為に必要なユーザをルートユーザにログインし、作成します。なお、ユーザの作成は、システムコマンドまたは、Vine Linuxの管理ツールより実施します。

以下コマンドでのユーザ作成方法について示します。

```
# adduser ユーザ名 -u ユーザ ID -g 初期グループ -G 追加グループ 1
-s /bin/csh (ログインシェルをCシェルに設定)
```

例) admuser の場合

```
# adduser admuser -u 500 -g webuser -G admusers -s /bin/csh
```

本システムでのユーザの設定を以下に示します。

```
admuser      : システム管理者用ユーザ
webuser      : WWWサーバ用ユーザ
```

ユーザ名	ユーザ ID	初期グループ	追加グループ
admuser	500	webuser	admusers
webuser	600	webuser	

尚、webuser の場合は、セキュリティの関連上、パスワードをロックすることを推奨します。

ロックするには、/etc/passwd ファイルを編集するか、あるいは、Vine Linux 管理者用ツールにて実施します。

以下に/etc/passwd の編集方法について記載します。
etc/passwd 上の webuser: の後に "*" を記入します。

例

```
webuser:*:600:501:,,,,:/home/webuser:/bin/csh
```

4 . 1 . 2 使用環境設定	(2) ベースソフトウェア環境設定	4/6
<p>(f) ネットワークの設定</p> <p>以下にネットワークの設定方法について記載します。</p> <p>(ア) root でログインします。</p> <p>(イ) Xwindow を立ち上げます。</p> <p style="padding-left: 2em;">% startx</p> <p>(ウ) ネットワークの設定を立ち上げます。</p> <p>(ア) マウスの右ボタンをクリックします。</p> <p>(イ) ユーティリティ 管理ツール コントロールパネルを選択します。</p> <p>(ウ) Network Configuration を選択します。</p> <p>(エ) Names パネルにて、Hostname、Domain、Nameservers を設定します。</p> <p>(オ) Hosts パネルにて、IP、Name、Nicknames を設定します。</p> <p>(カ) Interfaces パネルにて、Interface、IP、proto、atboot、active を設定します。</p> <p>(キ) Routing パネルにて、Default Gateway、Default Gateway Device を設定します。</p>		

MHI でのネットワークの設定例を以下に示します。

Names パネル

設定項目	設定内容	備考
Hostname	mhi170.ctc.ne.jp	サーバのホスト名
Domain	ctc.ne.jp	サーバのドメイン
Nameservers	210.158.0.17	DNS サーバ

Hosts パネル

設定項目	設定内容
IP	210.167.169.170
Name	mhi.170.ctc.ne.jp
Nicknames	mhi170

Interfaces パネル

設定項目	設定内容	
interface	lo	eth0
IP		210.167.169.170
proto	none	none
atboot		yes
atboot	active	active

Routing パネル

設定項目	設定内容
Default Gateway	210.167.169.161
Default Gateway Device	eth0

(B)webuser の環境を設定 (.cshrc ファイルを編集します)

.cshrc の内容を以下のように変更します。

```
% cd /home/webuser  
% vi .cshrc
```

記載する内容

```
# .cshrc
```

```
# User specific aliases and functions
```

```
setenv PATH "/home/wg71/tool:/home/wg71/pgm/bin:/usr/sbin:/sbin:${PATH}"  
setenv CLASSPATH ./usr/local/java/Apache-JServ/lib/ApacheJServ.jar:  
/usr/local/java/JSDK2.0/lib/jsdk.jar:  
/usr/local/java/xml-tr2/xml.jar:/usr/local/java/cos/lib/cos.jar:  
/usr/local/java/csv/csv.jar:/home/wg71/pgm  
set path=( $path /usr/local/java/jdk117_v3/bin /home/webuser )  
umask 7
```

(3) フリーソフトウェアのインストール及び設定

本システムが稼働するために必要な以下のベースソフトウェアのインストールおよび設定方法を記述します。

- (A) mirrorのインストール
- (B) Apacheのインストール
- (C) Open SSLのインストール
- (D) mod_sslのインストール
- (E) JDK及びshuJITのインストール
- (F) JSDKのインストール
- (G) Apache JServのインストール
- (H) Java ProjectX TR2のインストール
- (I) csv.jarのインストール
- (J) cos.jarのインストール
- (K) Apacheの設定
- (L) サイト秘密キーの設定
- (M) サイト証明書を作成
- (N) Apache起動時のパスフレーズ入力の自動
- (O) boot時の自動起動の設定
- (P) ベーシック認証のパスワードファイルの作成
- (Q) SSL用排他制御ディレクトリの作成

尚、各フリーソフトのダウンロード先を以下に示します。

ソフトウェア名称	ダウンロード先
Mirror 2.9	http://sunsite.org.uk/packages/mirror/
Apache 1.3.9	http://www.apache.org/
OpenSSL 0.9.4	http://www.openssl.org/
mod_ssl2.4.2	http://www.modssl.org/
JDK1.1.7	http://java.sun.com/j2se/
shuJIT 0.39	http://www.shudo.net/jit/dist/
JSDK2.0	http://java.sun.com/products/servlet/index.html
Apache Jserv 1.0	http://java.apache.org/
Java ProjectX TR2	http://java.sun.com/xml/
csv.jar	http://www.wakhok.ac.jp/~tomoharu/Java/csv/
cos.jar	http://www.amazon.com/exec/obidos/ASIN/156592391X/jasonhunter

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	1/24
<p>(A) mirrorのインストール 以下にmirrorの設定手順を示します。</p> <p>(a) 事前に perl がインストールされている事を確認します。 % ls /usr/bin/perl ・以下のメッセージが表示されれば perl がインストールされています。 /usr/bin/perl</p> <p>(b) root ユーザにてログインします。(または root ユーザにスイッチします)</p> <p>(c) インストールファイルを解凍します。 (今回は/usr/local/src/mirror-2.9 に解凍する例を示します。)</p> <pre>% cd /usr/local/src/mirror-2.9 (ディレクトリが存在しなかった場合は、ディレクトリを作成します。) % cp /copysrcpath/mirror.tar.gz ./ (copysrcpath はコピー元のパス) % tar -zxvf mirror.tar.gz</pre> <p>(d) インストールします。 (ア) インストーラーを実行します。 % su # perl install.pl here</p> <p>・実行すると以下のメッセージが表示されます。 About to install mirror To the following questions press return to use the default value shown in brackets or enter a new value.</p>		

(イ) perl の場所を入力します。

The command to run perl programs is [/usr/bin/perl]?

・ [] 内にデフォルトを表示しているのものでそのままの場合
Enter を押します。

・ Enter 押下後、以下のメッセージが表示されます。

```
using /usr/bin/perl
```

(ウ) インストール先のディレクトリを入力します。

(今回は、/usr/local/etc/mirror にインストールする例を示します。)

・ インストール先のディレクトリ確認メッセージが表示された後に、
インストール先ディレクトリを入力します。(下線が入力箇所)

the place to install mirror commands is

```
[/usr/local/src/mirror-2.9]?
```

```
/usr/local/etc/mirror
```

(エ) インストール先のディレクトリを作成します。

(今回は、/usr/local/etc/mirror にインストールする例を示します。)

・ インストール先のディレクトリを作成する確認メッセージが
表示された後に “ y ” を入力します。(下線が入力箇所)

```
Shall I create /usr/local/etc/mirror [n]? y
```

```
using /usr/local/etc/mirror
```

(オ) インストール先の再確認メッセージを確認します。

(今回はそのまま Enter を押します)

The place to install mirror support libraries [/usr/local/etc/mirror]?

```
using /usr/local/etc/mirror
```

```
Installing mirror
```

```
Install mirror.pl into /usr/local/etc/mirror/mirror
```

```
Install mm.pl into /usr/local/etc/mirror/mm
```

```
Install pkgs_to_mmin.pl into /usr/local/etc/mirror/pkgs_to_mmin
```

```
Install prune_logs.pl into /usr/local/etc/mirror/prune_logs
```

```
Install do_unlinks.pl into /usr/local/etc/mirror/do_unlinks
```

```
#
```

・ インストール完了

(e) インストール先にインストールファイルを展開したディレクトリから perl ファイル(*.pl)をコピーします。

```
# cp /usr/local/src/mirror-2.9/*.pl /usr/local/etc/mirror
```

(f) インストール先にインストールファイルを展開したディレクトリから mirror.defaults をコピーします。

```
# cp /usr/local/src/mirror-2.9/mirror.defaults
/usr/local/etc/mirror
```

(g) mirror.defaults を編集してミラーの設定を変更します。
以下に UNIX 標準エディタの vi にて編集する例を示します。

```
# cd /usr/local/etc/mirror
# vi mirror.defaults
```

設定内容については添付資料 5 を参照して下さい。

(h) mirror 先の設定をします。

(ア) ミラー先のサーバにミラー処理用のユーザを登録します。

注：ミラー先のユーザの登録内容は (g) にて編集します
mirror.defaults に反映しておきます。

(反映の記述については添付資料 0 を参照。)

MHI での設定例を以下に示します

設定項目	設定内容
ユーザ名	mirror
ユーザID	700
グループ名	mirror
グループID	700
パスワード	*****

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	4/24
<p>(イ) ミラー先のサーバにミラー処理用のディレクトリを作成します。 (今回は /usr/local/etc/mirror を作成する例を示します。)</p> <pre data-bbox="347 524 671 607">% cd /usr/local/etc/ % mkdir mirror</pre> <p>注：作成するミラー処理用のディレクトリは (h)(ア) にて登録した ユーザに read/write の権限があるディレクトリ内に作成します。</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	5/24
<p>(B) A p a c h e のインストール</p> <p>以下にWWWサーバであるA p a c h e の設定手順を示します。 尚、A p a c h e にはD S O 機能及び mod_rewrite (URL Rewriting Engine) も同時に組み込みます。</p> <p>(a) 既存の Apache を停止します。(root ユーザでのみ実行可能)</p> <pre>% /etc/rc.d/init.d/httpd stop</pre> <p>(b) インストールファイルを解凍します。 (/usr/local/src に解凍する例を示します)</p> <pre>% cd /usr/local/src % cp /copysrcpath/apache_1.3.9.tar.gz ./ (copysrcpath はコピー元のパス) % gzip -cd apache_1.3.9.tar.gz tar xvf - (解凍後 apache_1.3.9 ディレクトリが作成されます) % cd apache_1.3.9</pre> <p>(c) Apache の構成を設定します。 (/usr/local/apache にインストールする例を示します)</p> <pre>% ./configure --prefix=/usr/local/apache --enable-rule=SHARED_ CORE --enable-module=so --enable-module=rewrite --with-perl=/usr/bin</pre> <p>注 1 : 別のディレクトリにパスが通っていることがあるので ./ を必ず指定します</p> <p>注 2 : --enable で始まる部分は DSO 及び mod_rewrite 組み込みの為の指定</p> <p>注 3 : 今回は /usr/bin の下に perl ファイルが存在しました</p> <p>(d) モジュールを組み込む。</p> <p>(b) で src ディレクトリに Configuration.apaci が作成されるのでこれをコピーしてモジュールを組み込みます。</p> <pre>% cd src % cp Configuration.apaci Configuration</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	6/24
<p>(e) 組み込んだモジュールをソースに反映させます。 (src ディレクトリで実行します)</p> <pre data-bbox="347 479 560 510">% ./Configure</pre> <p>注 : 別のディレクトリにパスが通っていることがあるので ./ を必ず指定します</p> <p>(f) コンパイルします。(apache_1.3.9 ディレクトリで実行します)</p> <pre data-bbox="347 770 887 936">% cd .. (apache_1.3.9 ディレクトリへ移動) % make clean % make</pre> <p>(g) インストールします。(root 権限で実施します)</p> <pre data-bbox="347 1106 571 1182">% su # make install</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	7/24
<p>(C) OpenSSLのインストール</p> <p>(a) インストールファイルを解凍します。 (今回は/usr/local/src/Apche-SSLに解凍します)</p> <pre>% cd /usr/local/src/Apche-SSL % cp /copysrcpath/openssl-0.9.4.tar.gz ./ (copysrcpathはコピー元のパス) % gzip -cd openssl-0.9.4.tar.gz tar xvf - (解凍後openssl-0.9.4ディレクトリが作成されます) % cd openssl-0.9.4</pre> <p>(b) opensslの構成を設定します。(今回はデフォルトにインストールします)</p> <pre>% ./config</pre> <p>(注) 非常に時間がかかります インストール先を指定する場合は以下の通りです。</p> <pre>% ./config --prefix=installpath --openssldir=meltpath</pre> <p>注1：別のディレクトリにパスが通っていることがあるので./を必ず指定します 注2：installpath(インストール先パス)、meltpath((1)で解凍したパス)</p> <p>(c) コンパイルします。</p> <pre>% make % make test</pre> <p>(d) インストールします。(root権限で実施します)</p> <pre>% su # make install</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	8/24
<p>(D) mod_ssl のインストール</p> <p>(a) インストールファイルを解凍します。 (今回は/usr/local/src/Apache-SSL に解凍します)</p> <pre>% cd /usr/local/src/Apache-SSL % cp /copysrcpath/mod_ssl-2.4.2-1.3.9.tar.gz ./ (copysrcpath はコピー元のパス) % gzip -cd mod_ssl-2.4.2-1.3.9.tar.gz tar xvf - (解凍後 mod_ssl-2.4.2-1.3.9 ディレクトリが作成されます) % cd mod_ssl-2.4.2-1.3.9</pre> <p>(b) apache の構成を設定します。</p> <pre>% ./configure --with-apache=../../apache_1.3.9 --with-ssl= ../openssl-0.9.4 --prefix=/usr/local/apache --enable-shared=ssl</pre> <p>注1 : 別のディレクトリにパスが通っていることがあるので./を必ず指定します</p> <p>注2 : --with-apache には apache のソースディレクトリを指定します (今回は../../apache_1.3.9) --with-ssl には openssl のソースディレクトリを指定します (今回は../openssl-0.9.4) --prefix には apache インストールディレクトリを指定します (今回は/usr/local/apache)</p> <p>(c) apache の再コンパイルをします。(apache の build ディレクトリで実施します)</p> <pre>% cd ../../apache_1.3.9 % make</pre> <p>(d) インストールします。(root 権限で実施します)</p> <pre>% su # make install</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	9/24
<p>(E) J D K 及び s h u J I T のインストール</p> <p>(a) インストールファイルを解凍します。(今回は/usr/local/java に解凍します)</p> <pre data-bbox="347 571 1244 940">% cd /usr/local/java % cp /copysrcpath/jdk_1.1.7-v3-glibc-x86.tar.gz ./ % cp /copysrcpath/ shujit-0.3.9-bin-linux-glibc-jdk11.tar.gz ./ (copysrcpath はコピー元のパス) % gzip -cd jdk_1.1.7-v3-glibc-x86.tar.gz tar xvf - (解凍後 jdk117_v3 ディレクトリが作成される) % gzip -cd shujit-0.3.9-bin-linux-glibc-jdk11.tar.gz tar xvf - (解凍後 shujit ディレクトリが作成される)</pre> <p>(b) shuJIT をライブラリディレクトリにコピーします。 (今回は/usr/local/java/jdk117_v3/lib/i686/green_threads/にコピーします)</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	10/24
<p>(F) J S D K のインストール</p> <p>(a) インストールファイルを解凍します。</p> <p>(今回は /usr/local/java に解凍します)</p> <pre>% cd /usr/local/java % cp /copysrcpath/jsdk20-solaris2-sparc.tar.Z ./ (copysrcpath はコピー元のパス) % tar xfvz jsdk20-solaris2-sparc.tar.Z (解凍後 JSDK2.0 ディレクトリが作成されます)</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	11/24
<p>(G) Apache JServ のインストール</p> <p>(a) インストールファイルを解凍します。 (今回は /usr/local/java に解凍します)</p> <pre>% cd /usr/local/java % cp /copysrcpath/Apache_JServ_1.0.tar.gz ./ (copysrcpath はコピー元のパス) % gzip -cd Apache_JServ_1.0.tar.gz tar xvf - (解凍後 ApacheJServ-1.0 ディレクトリが作成されます) % cd ApacheJServ-1.0</pre> <p>(b) 英語版 Perl をインストールします。 (英語版でないとコンパイルが通らないためです)</p> <pre>% rpm -e --nodeps perl (日本語版のアンインストール) 注 1 : 「 を削除できません。」というメッセージが表示されるが 問題はありません。 % cd pealdir (pealdir は英語版 Perl が保管してあるディレクトリです) % rpm -i perl-5.004m7-1.i386.rpm (英語版のインストール)</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	12/24
<p>(c) Apache JServ の構成を設定します。</p> <pre data-bbox="347 427 1428 656">% ./configure --with-apache-install= /usr/local/apache --prefix=/usr/local/java/Apache-Jserv--with-jdk- home=/usr/local/java/jdk117_v3 --with-jsdk= /usr/local/java/JSDK2.0/lib/jsdk.jar --enable-compressed -jar--enable-debugging</pre> <p>注 1 : 別のディレクトリにパスが通っていることがあるので ./ を必ず指定します</p> <p>注 2 : --with-apache-install には apache のインストールディレクトリを指定します (今回は /usr/local/apache) --prefix には Apache JServ のインストールディレクトリを指定しま</p> <p>す</p> <p>(今回は /usr/local/java/Apache-Jserv) --with-jdk-home には jdk の解凍ディレクトリを指定します (今回は /usr/local/java/jdk117_v3) --with-jsdk には jsdk のクラスパスを指定します (今回は /usr/local/java/JSDK2.0/lib/jsdk.jar)</p> <p>(d) コンパイルします。</p> <pre data-bbox="347 1339 448 1368">% make</pre> <p>(e) インストールします。(root 権限で実施します)</p> <pre data-bbox="347 1485 571 1563">% su # make install</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	13/24
<p>(f) Perl を日本語版に戻します。</p> <pre>% rpm -e --nodeps perl</pre> <p>(英語版のアンインストール)</p> <p>注 1 : 「 を削除できません。」というメッセージが表示されるが問題はありません。</p> <p>Vine Linux1.1CR の CD-ROM を入れます。</p> <pre>% mount /dev/cdrom /mnt/cdrom</pre> <pre>% cd /mnt/cdrom/Vine/RPMS</pre> <pre>% rpm -i perl-5.004_jp-3.i386.rpm</pre> <p>(日本語版のインストール)</p> <pre>% cd /</pre> <pre>% umount /mnt/cdrom</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	14/24
<p>(H) J a v a P r o j e c t X T R 2 のインストール</p> <p>(a) インストールファイルを解凍します。 (今回は/usr/local/java に解凍します)</p> <pre>% cd /usr/local/java</pre> <pre>% cp /copysrcpath/xml-tr2.zip ./</pre> <p>(copysrcpath はコピー元のパス)</p> <pre>% unzip xml-tr2.zip</pre> <p>(解凍後 xml-tr2 ディレクトリが作成されます)</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	15/24
<p>(I) csv.jarのインストール (今回は/usr/local/java/csvにコピーします)</p> <pre>% cd /usr/local/java % mkdir csv % cd csv % cp /copysrcpath/csv.jar ./ (copysrcpath はコピー元のパス)</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	16/24
<p>(J) <code>cos.jar</code> のインストール (今回は <code>/usr/local/java</code> に解凍します)</p> <pre>% cd /usr/local/java % mkdir cos % cd cos % cp /copysrcpath/cos.zip ./ (copysrcpath はコピー元のパス) % unzip cos.zip</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	17/24
<p>(K) A p a c h e の設定 (コンフィグファイルの設定)</p> <p>(a) Apache の httpd.conf ファイルを設定します。 以下に UNIX 標準エディタの vi にて編集する例を示します。 <pre>% cd /usr/local/apache/conf % su # vi httpd.conf</pre> 設定内容については添付資料 1 を参照してください 注 1 : 設定ファイルは httpd.conf と srm.conf と access.conf の 3 つあります がデフォルトでは httpd.conf に srm.conf と access.conf の内容が含ま れているので httpd.conf のみを変更します。</p> <p>(b) Apache Jserv の httpd.conf ファイルを設定します。 以下に UNIX 標準エディタの vi にて編集する例を示します。 <pre>% cd /usr/local/java/ApacheJServ-1.0/conf % su # vi httpd.conf</pre> 設定内容については添付資料 2 を参照してください。</p> <p>(c) Apache Jserv の jserv.properties ファイルを設定します。 以下に UNIX 標準エディタの vi にて編集する例を示します。 <pre>% cd /usr/local/java/ApacheJServ-1.0/conf % su # vi jserv.properties</pre> 設定内容については添付資料 3 を参照してください</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	18/24
<p>(d) Apache Jserv の zone.properties ファイルを設定します。</p> <pre data-bbox="347 427 1075 896">% cd /usr/local/java/ApacheJServ-1.0/conf % su # cp zone.properties share_zone.properties # vi share_zone.properties # cp zone.properties secure_zone.properties # vi secure_zone.properties # cp zone.properties system_zone.properties # vi system_zone.properties # cp zone.properties customer_zone.properties # vi customer _zone.properties</pre> <p>設定内容については添付資料 4 を参照してください</p> <p>注 1 : zone.properties ファイルは 4 ファイル作成します。(share 用、secure 用、system 用、customer 用)</p> <p>(それぞれの servlet を独立させ、セキュリティを確保するためです)</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	19/24
<p>(L) サイト秘密キーの設定</p> <p>(a) キーを置くディレクトリを作成します。 (今回は/usr/local/apache/etc/ssl/keysを作成します)</p> <pre>% su # mkdir /usr/local/apache/etc/ssl/keys # chmod 0700 /usr/local/apache/etc/ssl/keys # cd /usr/local/apache/etc/ssl/keys</pre> <p>ファイルを5つ用意します。 (適当なファイルを5つ/usr/local/apache/etc/ssl/keysへコピーします)</p> <pre># cp file ./</pre> <p>秘密キーを作成します。(今回はf1、f2、f3、f4、f5の5ファイルを用意した)</p> <pre># /usr/local/ssl/bin/openssl genrsa -des3 -rand f1:f2:f3:f4:f5 1024 > httpd.key</pre> <p>パスフレーズをきいてくるのでパスフレーズを入力します(確認のため2回) 注：/usr/local/ssl/binはopensslをインストールしたディレクトリ</p> <p>(b) 作成されたキーhttpd.keyの権限を変更します。</p> <pre># chmod 0400 httpd.key</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	20/24
<p>(M) サイト証明書の作成</p> <p>(a) 証明書を置くディレクトリを作成します。 (今回は/usr/local/apache/etc/ssl/certsを作成します)</p> <pre>% su # mkdir /usr/local/apache/etc/ssl/certs # chmod 0700 /usr/local/apache/etc/ssl/certs # cd /usr/local/apache/etc/ssl/certs</pre> <p>(b) 証明書を作成します。</p> <pre># /usr/local/ssl/bin/openssl req -new -x509 -days 365 - key ../keys/httpd.key > httpd.pem</pre> <p>パスフレーズをきいてくるので秘密キーを作成したときのパスフレーズを入力します。</p> <p>その他に国、都道府県、市、組織、部署、名前、メールアドレスを入力します。</p> <p>注：/usr/local/ssl/bin は openssl をインストールしたディレクトリ</p> <p>(c) 作成された証明書 httpd.pem の権限を変更します。</p> <pre># chmod 0400 httpd.pem</pre>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	21/24
<p>(N) A p a c h e 起動時のパスフレーズ入力の自動化 デフォルトでは Apache の起動時にパスフレーズを要求してくるのでパスフレーズをプログラムで入力するようにします。</p> <p>(a) プログラムを作成します。(今回は /usr/local/apache/bin/pp-filter)</p> <p>プログラム(シェル)の内容は以下の通りです。</p> <pre data-bbox="347 712 863 846">#/usr/local/apache/bin/pp-filter #!/bin/sh echo パスフレーズ</pre> <p>注：このプログラムは root 権限でのみ実行できるようにしておきます。</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	22/24
<p>(O) b o o t 時の自動起動の設定</p> <p>(a) Vine Linux にインストールされていた Apache を削除します。</p> <pre># cd /usr/sbin # rm httpd</pre> <p>(b) 新しくインストールした Apache をコピーします。</p> <pre># cp /usr/local/apache/bin/httpd ./</pre> <p>注 : /usr/local/apache は apache インストールディレクトリ</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアのインストール及び設定	23/24
<p>(P) ベーシック認証のパスワードファイルの作成 (ここで設定したユーザとパスワードで管理者用ポートにアクセスできるようになります。)</p> <p>(a) パスワードファイルを作成します。</p> <pre>% cd /usr/local/apache/auth % su # htpasswd -c authuser_system ユーザ名 ユーザのパスワードを聞いてくるのでパスワードを入力します。 authuser_system ファイルが作成されます。</pre> <p>注 1 : -c はファイルを新規作成するのでユーザの追加のときは-c をとります。</p>		

4 . 1 . 2 使用環境設定	(3) フリーソフトウェアの インストール及び設定	24/24
<p>(Q) S S L 用排他制御ディレクトリの作成</p> <p>(a) /usr/local/apache/var/run ディレクトリを作成します。</p> <pre data-bbox="347 571 703 750">% cd /usr/local/apache % mkdir var % cd var % mkdir run</pre>		

(4) インターネットへの接続確認

設定したサーバがインターネット上から参照可能であることの確認方法を記述します。

4 . 1 . 2 使用環境設定	(4) インターネットへの接続確認	1/1
<p>(A) インターネット上の URL 名を検索します。</p> <p>(a) “nslookup”コマンドを実行すると以下のメッセージが表示されます。</p> <p>例</p> <pre>% nslookup Default Server: ns2.ctc.ad.jp DNS サーバ名が表示されます。 Address: 210.158.0.17 DNS の I P アドレスが表示されます。</pre> <p>(b) インターネット上の URL 名を入力します。</p> <p>例 :</p> <pre>> mhi170.ctc.ne.jp</pre> <p>(c) 以下のメッセージが表示されることを確認します。</p> <p>例 :</p> <pre>Server: ns2.ctc.ad.jp Address: 210.158.0.17 Name: mhi170.ctc.ne.jp 入力したインターネット上の URL 名 Address: 210.167.169.170 入力したインターネット上の I P アドレス</pre> <ul style="list-style-type: none"> ・ “Name: インターネット URL 名”が表示されればインターネットが接続されている 		

(5) フリーソフトウェアの稼働確認

以下のフリーソフトウェアが正常に稼働することの確認方法を記述します。

- (A) `mirror`の稼働確認
- (B) `Apache`の稼働確認
- (C) `Jserv`の稼働確認

(A) mirrorの稼働確認

(a) mirrorを起動します (root 権限で起動します)

```
% su  
# ./mirror -d
```

注1 : 実行時に XX.ph が無いというエラーが出た場合は以下の処置をります。
(今回は stddef.ph が無かった場合の例を示します。)

(ア) stddef.ph を検索します。

```
# find / -name stddef.ph -print
```

・ stddef.ph が検索された場合は所定の場所へコピーします。

```
# cp /usr/lib/perl5/i386-linux/5.00404/linux/stddef.ph /usr/lib/  
perl5/i386-linux/5.00404
```

・ stddef.ph が見つからない場合は stddef.h を検索し、perl のヘッダ
ファイルを作成します。(root 権限で起動します)

(今回は stddef.h が /usr/include/gnu に有った場合の例を示します。)

```
# cd /usr/include; h2ph gnu/*.h
```

(b) パッケージ指定で実行します。(root 権限で起動します)

(今回は package 名を edimaster とした例を示します。)

```
% su  
# ./mirror -pedimaster
```

(c) 設定通りにディレクトリがミラー先のサーバにミラーリングされて
いるか確認します。

4 . 1 . 2 使用環境設定	(5) フリーソフトウェアの稼働確認	2/3
<p>(B) A p a c h e の稼働確認</p> <p>(a) Apache を起動します。(root 権限で起動します)</p> <pre data-bbox="347 524 976 607">% su # /usr/local/apache/bin/apachectl start</pre> <p>注 1 : /usr/local/apache は“(B) Apache のインストール - (b) ”で指定したディレクトリ</p> <p>注 2 : bin ディレクトリは“(B) Apache のインストール - (b) ”でディレクトリ構成の確認をしたときの sbindir のパス</p> <p>(b) ブラウザで初期画面が表示されることを確認します。 任意のブラウザを起動します。 ブラウザ上の http コマンドラインより以下のコマンドを実行します。 http://サーバの URL 名 (E D I サーバに設定した URL 名を入力します。) 例 : http://mhi170.ctc.ne.jp ブラウザ上に Apache の初期画面が表示されることを確認します。</p> <p>(c) S S L 通信の確認 セキュア用 URL をアドレス欄から入力し、サーバ認証キーが正常に表示されることを確認します。</p> <p>アドレス : https://サーバの URL 名</p> <p>例 : https://mhi170.ctc.ne.jp</p>		

(C) J s e r v の稼働確認

(a) Apache を起動します。(root 権限で起動します)

```
% su
```

```
# /usr/local/apache/bin/apachectl start
```

エラーが出る場合はコンフィグファイル等の設定を確認します。

注 1 : Apache を起動すると自動で Apache-Jserv が起動する設定になっています。

(b) Apache-Jserv が起動しているか確認します。

```
% lynx http://localhost/jserv/
```

Apache-JServ が稼働していれば、"ApacheJServ-1.0 Status" ページが表示されます。

エラーが出る場合はコンフィグファイル等の設定を確認します。

(c) Servlet が正常に動作するかを確認します。

```
% cd /usr/local/java/ApacheJServ-1.0/example
```

```
% cp Hello.class /home/wg73/share/servlets/
```

ブラウザから、下記の URL を表示させます。

```
http://<hostname>/servlets/Hello
```

Servlet が正常に動作している場合は

```
Example Apache JServ Servlet  
Congratulations, Apache JServ is working!
```

の画面が表示されます。

4 . 1 . 3 社内システムとのインターフェース

ここでは、発注側 EDI サーバと社内システムとのインターフェースについて説明します。

- ・ 社内システム EDI サーバ (1)参照
- ・ EDI サーバ 社内システム (2)参照

(1) 社内システム EDI サーバ

発注側社内システムで作成した受発注データを EDI サーバへ送り込む際のインターフェースについて記述します。

4.1.3 社内システムとの インターフェース	(1) 社内システム EDI サーバ	1/3
----------------------------	--------------------	-----

(A) EDI データファイル

(a) ディレクトリ

edi_server_env.xml で指定してあるディレクトリに FTP 等でファイルを送信します。(デフォルトは/home/wg71/edi/inner/input/)

(b) ファイル名

名前が重複しない、拡張子 xml

(c) ファイル形式

XML 形式で該当する情報区分の DTD に沿った文書構造とし、情報区分が異なる場合、または運用モードが異なる場合にはファイルを分割し、異なる情報区分や、異なる運用モードが1ファイルに混在してはいけません。

MGH (メッセージグループヘッダー) の運用モード、情報区分コードは必須入力とします。

XML 文書の DOCTYPE 宣言を必ず付け、DTD はカレントを参照するようにしてください。

DOCTYPE 宣言の例を以下の表に示します。

情報区分 コード	情報区分	DOCTYPE 宣言
0301	見積依頼情報	<? DOCTYPE SJAC0301 SYSTEM "SJAC0301.dtd" ?>
0303	円貨確定依頼情報	<? DOCTYPE SJAC0303 SYSTEM "SJAC0303.dtd" ?>
0500	注文情報	<? DOCTYPE SJAC0500 SYSTEM "SJAC0500.dtd" ?>
0601	注文残高情報	<? DOCTYPE SJAC0601 SYSTEM "SJAC0601.dtd" ?>
0701	納期確認情報	<? DOCTYPE SJAC0701 SYSTEM "SJAC0701.dtd" ?>
0705	入荷情報	<? DOCTYPE SJAC0705 SYSTEM "SJAC0705.dtd" ?>
0801	検査情報	<? DOCTYPE SJAC0801 SYSTEM "SJAC0801.dtd" ?>
0901	検収情報	<? DOCTYPE SJAC0901 SYSTEM "SJAC0901.dtd" ?>
1101	買掛明細情報	<? DOCTYPE SJAC1101 SYSTEM "SJAC1101.dtd" ?>

(d) 文字コード

文字コードは UTF-8 にて作成してください。

4.1.3 社内システムとの インターフェース	(1) 社内システム EDI サーバ	2/3
<p>(e) 権限 ファイルには apache で設定してあるユーザ（デフォルトでは webuser）で削除可能な権限を与えることとします。（詳細は添付資料 1 を参照）</p> <p>(f) 処理順 EDI サーバはファイルの作成順に処理します。</p> <p>(g) エラー処理 エラーチェックはファイル単位に実施されます。ファイル内の 1 メッセージでエラーが発生した場合、エラーメッセージのみを取込まないのではなく、ファイル内のすべてのメッセージを取込みません。</p>		

4 . 1 . 3 社内システムとの インターフェース	(1) 社内システム EDI サーバ	3/3
<p>(B) 削除指示ファイル</p> <p>(a) ディレクトリ edi_server_env.xml で指定してあるディレクトリに FTP 等でファイルを送信します。(デフォルトは/home/wg71/edi/inner/input/)</p> <p>(b) ファイル名 YYYYMMDDhhmmss-情報区分コード (4 桁) -deletemessage.xml (YYYYMMDDhhmmss は作成した年月日時分秒ファイル名が重複しないように定めます。)</p> <p>(c) ファイル形式 XML 形式で構造仕様書のファイル仕様の削除指示ファイルの仕様に沿った文書構造とし、情報区分コード毎に 1 ファイルとします。 識別キーの受注者コードは必須入力とし、他の識別キーはブランクならばすべてとして扱こととします。</p> <p>(d) 権限 ファイルには apache で設定してあるユーザ (デフォルトでは webuser) で削除可能な権限を与えることとします。(詳細は添付資料 1 を参照)</p> <p>(e) 処理順 EDI サーバはファイルの作成順 (EDI サーバにファイルが作成された日付順) に処理します。</p>		

(2) EDI サーバ 社内システム

受注側より送信された受発注データを EDI サーバより抽出し、発注側社内システムへ取り込む際のインターフェースについて記述します。

4 . 1 . 3 社内システムとの インターフェース	(2) EDI サーバ 社内システム	1/1
--------------------------------	----------------------	-----

(A) EDI データファイル

(a) ディレクトリ

edi_server_env.xml で指定してあるディレクトリにファイルが作成されます。(デフォルトは/home/wg71/edi/inner/output/)

(b) ファイル名

YYYYMMDDhhmmss-情報区分コード(4桁)-運用モード(1桁).xml
(YYYYMMDDhhmmss は年月日時分秒)

(c) ファイル形式

XML 形式で該当する情報区分の DTD に沿った文書構造とし、情報区分コード・運用モード毎に1ファイルとなります。

XML 文書の DOCTYPE 宣言が付き、DTD はカレントを参照します。

作成される DOCTYPE 宣言を以下の表に示します。

情報区分 コード	情報区分	DOCTYPE 宣言
0302	見積回答情報	<? DOCTYPE SJAC0302 SYSTEM "SJAC0302.dtd" ?>
0304	円貨確定情報	<? DOCTYPE SJAC0304 SYSTEM "SJAC0304.dtd" ?>
0702	納期回答情報	<? DOCTYPE SJAC0702 SYSTEM "SJAC0702.dtd" ?>
0704	出荷情報	<? DOCTYPE SJAC0704 SYSTEM "SJAC0704.dtd" ?>

(d) 文字コード

文字コードは UTF-8 にて作成されます。

(e) 権限

ファイルには apache で設定してあるユーザ(デフォルトでは webuser)で作成されます。(詳細は添付資料1を参照)